University of Ibadan
Journal of Science and Logics in ICT Research

Enhancing Symmetric Encryption Using Digital Signatures

✉**Chamo Y. Y.**
Department of Computer Science, Faculty of Computing, National Open University of Nigeria, Abuja.
yusufchamo@gmail.com

Abstract
Maintaining the confidentiality and integrity of digital documents transmitted through electronic media is a critical security concern in the field of Information Security. To address this security concern, this paper proposes a system that uses a digital signature to ensure the authenticity, non-repudiation and integrity of the transmitted data and it also uses symmetric encryption to provide authentication and confidentiality of the transmitted data. The Rivest, Shamir & Adleman (RSA) algorithm was used to implement the Digital Signature while the Advanced Encryption Standard (AES) was used for symmetric encryption. The system involves encrypting a plaintext using AES, then a hash function (SHA-256) is used to create a hash value of the ciphertext and the private key of the RSA algorithm is used to encrypt the hash value to produce the digital signature. The ciphertext and the digital signature are attached and sent to the recipient. The digital signature is decrypted by the recipient to obtain the hash value of the ciphertext, then it verifies if it is a valid signature before proceeding to decrypt the ciphertext using the AES secret key. The proposed system was evaluated against the existing AES algorithm. The size of the test file was observed and analyzed before and after encryption, this showed that the size did not change. Different RSA key sizes were used to perform signature and verification processes to see how long it takes to perform the operations, this also showed that the smaller the key size the faster the signature and verification processes and the verification process is a much faster process than the signature process. The system was able to meet the cryptography objectives and will be useful to individuals and businesses in transmitting sensitive information over insecure communication mediums.

Keywords*: Advanced Encryption Standard (AES); Rivest-Shamir-Adleman (RSA); Digital Signature; Encryption; Hash Function*

1.0 Introduction

In the digital age, the need for secure and trusted electronic transactions has grown exponentially. This paper highlights the fundamental role played by digital signatures in addressing this need. These cryptographic tools, rooted in a history of pioneering work by Diffie, Hellman, and the creators of the RSA algorithm, have evolved into indispensable components of modern cybersecurity. Their capacity to guarantee the integrity and non-repudiation of digital documents has made them a linchpin in the transition from paper-based transactions to digital transactions.

Cryptographic methods require copyright protection of digital documents and data as it only ensure data security during the distribution process [27]. This paper combines the strengths of symmetric encryption which are confidentiality and authentication of sensitive documents and the strengths of digital signature which are ensuring the integrity and non-repudiation of digital documents. The concept of using cryptographic techniques to ensure document authenticity can be traced back to the work of Whitfield Diffie and Martin Hellman, who introduced the concept of public-key cryptography in the 1970s [7]. This breakthrough laid the foundation for modern digital signatures. Later, in 1977, Ron Rivest, Adi Shamir, and Leonard Adleman developed the RSA algorithm, one of the earliest and most widely used digital signature algorithms [21]. Over the years, digital signature technologies have continued to evolve, accommodating advances in computing power and cryptographic research.

Document content summary can be done by using a hash function that produces an output called a hash value, summary encoding and encrypted summary insertion are the main digital signature processes [19] [27]. Digital signatures are based on asymmetric encryption techniques and involve the use of public and private key pairs [12]. The process begins with the sender

Chamo Y. Y. Enhancing Symmetric Encryption Using Digital Signatures. *University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR),* Vol. 13 No. 1, pp. 179 – 187.

using a cryptographic hash function to create a unique hash value of the document. The hash value is encrypted using a private key to form the digital signature. The recipient decrypts the digital signature using a public key. To verify the authenticity and integrity of the document, the recalculated hash value of the received document is compared against the decrypted hash value to see if they match.

Digital signatures offer several advantages in file encryption systems. Firstly, they provide a high level of integrity ensuring that no alteration of the document occurs during transmission. Secondly, non-repudiation of digital signatures ensures non-denial of signing the document by the sender. Thirdly, digital signatures enhance the authenticity of the document by verifying the identity of the sender through the use of the sender's private key. These advantages make digital signatures an essential component of secure file encryption systems [9].

Digital signatures find applications in various domains where secure communication and document authentication are critical. In e-commerce, digital signatures are used to authenticate electronic transactions, providing confidence to buyers and sellers [5]. In government and legal sectors, digital signatures are employed to sign legal documents, contracts, and agreements, eliminating the need for physical signatures. Adoption of digital signatures was accelerated during the COVID-19 pandemic as businesses and organizations increasingly rely on remote and digital processes [1].

One of the major challenges of symmetric encryption is the issue of security functions such as user verification and key management [15]. It is also not convenient for sharing sensitive information over insecure channels. This work proposed a system that combines the strengths of symmetric and digital signature algorithms to address this challenge. Digital signature was introduced to overcome the limitations of symmetric encryption and accomplish user verification, and to also ensure the integrity and authenticity of transmitted data [24].

2.0 Related Works

Aufa *et al.* [2] carried out a security system analysis by combining RSA and DSA (Digital Signature Algorithm) for encryption and digital signatures, which are both public key cryptography algorithms. They compared the computational times of their system against the RSA and DSA algorithms respectively. This work showed that their system provided three-way security of authentication, verification and data security but their system cannot also process large files due to the limitations of the RSA algorithm.

Lee and Kim [13] proposed an optimized approach that utilized efficient RSA key generation and encryption techniques for secure file encryption with digital signatures. This study demonstrated the effectiveness of the RSA algorithm in providing robust encryption for file-level data security and the ability to create digital signatures for file authentication. Their work was limited to processing small files.

Dhiyaulhaq & Usman [6] researched to find an optimal algorithm for the performance of digital signature by comparing the performance of the RSA 512-bit and AES 192-bit algorithms. The study showed that RSA has a more stable performance when carrying out digital signature operations compared to AES.

Bobby & Usha [3] carried out an Observational Study of Security Enhancement in IoT Authentication Using a Digital Signature Algorithm. They stated that the information exchanged in an IoT system is private and at times confidential, providing proper security to the system is essential. To prevent unknown users from accessing information in the system, authentication through Digital Signatures becomes an integral part of IoT. They also noted that Authentication plays an important role in internet security. Using Digital signatures, the authentication process is carried out well.

Lu & Mohammed [15] proposed a complex encryption system design which was implemented using AES and RSA. They noted that to analyze the performance of their proposed algorithm and to make full use of the advantages of AES, one needs to reduce round key and improve the key schedule, as well as organically integrate with the RSA algorithm. Java language was used to implement the proposed algorithm due to its large library. Based on the results of the comparison between AES and the proposed algorithm, the proposed algorithm showed good performance and high security. Their work was limited to using RSA to solve difficult key

management of RSA, as it did not incorporate digital signature operations.

William *et al.* [25] described and proposed a hybrid approach that combines a symmetric algorithm Advanced encryption standard (AES), asymmetric algorithm Elliptic curve cryptography (ECC), and a hash function (SHA-256). SHA-256 is a mathematical formula that is used to ensure the data's integrity. They noted that their proposed technique is more efficient when compared to the other approaches since it is more efficient when it comes to text encryption. A limitation of their proposed technique is that it is less efficient at image encryption. They also indicated that the time required to encrypt and decode a picture may be decreased in future development.

2.1 Digital Signature

Digital signatures rely on the principles of public-key cryptography. This is based on using two distinct but mathematically related keys: a public key and a private key; the former for encryption and the latter for decryption [7]. This fundamental principle ensures that while the public key is widely disseminated and accessible to anyone, only the possessor of the private key can create a valid digital signature. When signing a document, the sender uses their private key to apply a cryptographic transformation to the document, producing a signature. The senders' public key is used to verify the signature's authenticity by applying the corresponding inverse transformation. A successful verification guarantees the document's integrity & the identity of the sender. Digital signatures may get permanently linked to signed message content; but it is impossible to move them from document to document [20] [23].

Digital signature algorithms are the mathematical constructs that underpin the security and effectiveness of digital signatures. These algorithms generate the signature and ensure its integrity, making them a fundamental component of digital signature technology [18]. There are several notable digital signature algorithms, each with its unique characteristics, strengths, and weaknesses [26]. An example is the RSA, which was used to implement the digital signature in this paper. The choice of algorithm depends on the specific use case, security requirements, and performance considerations. Each algorithm has its own set of advantages and trade-offs, and ongoing research continues to improve and adapt digital signature algorithms to evolving security challenges.

Digital signatures find widespread use in various domains due to their ability to provide security, non-repudiation, and data integrity in the digital realm. Their versatility and reliability have led to their adoption in a range of practical applications. Here are some key areas where digital signatures play a crucial role:

i. **Secure Document Signing:** Digital signatures are commonly used to sign electronic documents, such as contracts, agreements, and legal paperwork. This ensures the authenticity and integrity of the documents, making them legally binding [4].
ii. **Email Authentication:** In the context of email, digital signatures help verify the sender's identity and the integrity of the message. This is vital for preventing email spoofing and ensuring secure communication.
iii. **E-commerce and Online Transactions:** Digital signatures are employed to secure online transactions, including purchases, financial transactions, and the signing of electronic checks. They ensure that the parties involved can trust the authenticity and integrity of the transactions [11].
iv. **Government and Identity Verification:** Governments use digital signatures for identity verification, issuing secure documents like passports, driver's licenses, and national ID cards. This enhances security and prevents fraud.
v. **Healthcare and Medical Records:** In the healthcare sector, digital signatures are applied to ensure the integrity and confidentiality of patient records, prescriptions, and medical documents. They also support telemedicine and remote healthcare services.
vi. **Cryptocurrency and Blockchain Technology:** Digital currencies, like Bitcoin, use digital signatures to validate transactions and ensure the security of cryptocurrency wallets. Blockchain technology relies on digital signatures to record and verify transactions on a distributed ledger [17].
vii. **Software Updates and Code Integrity:** Software developers use digital signatures to sign their code and software updates. This guarantees that the software has not been

tampered with and helps users verify its authenticity.

viii. **Legal and Notary Services:** Digital signatures are utilized in the legal and notary services industry to authenticate legal documents, wills, and affidavits. This eliminates the need for physical presence while ensuring the legal validity of documents.

ix. **Remote Work and Telecommuting:** The advent of remote work and telecommuting has heightened the importance of digital signatures for remote document signing and secure virtual collaboration.

x. **Supply Chain and Logistics:** Digital signatures are used in supply chain management to verify the authenticity and integrity of shipping documents, reducing the risk of fraud and ensuring the smooth flow of goods.

2.2 RSA (Rivest, Shamir & Adleman) Algorithm

This is one of the most renowned digital signature algorithms that was introduced in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman. Based on their findings, they developed a system of two-key cryptography that would make it possible for two messages to transit safely via an unsecured channel without disclosing a secret key [29]. RSA relies on the mathematical properties of large prime numbers. It allows for the creation of key pairs: public and private keys; for verification and signing respectively [10]. When a user signs a document, the algorithm employs modular exponentiation to create the digital signature. RSA is known for its security and robustness, and it remains popular for various applications.

2.2.1 RSA Key Generation

The algorithms used in generating RSA public key pairs are stated below [16]:

1. Two random prime numbers p and q are chosen such that the bit length of p is approximately equal to the bit length of q;
2. Calculate n so that $n = p * q$;
3. Calculate $\varphi(n)$ so that $\varphi(n) = (p - 1)*(q - 1)$;
4. Choose a random integer e such that $e< \varphi(n)$ and $gad(e, \varphi(n)) = 1$, then compute the integer d such that: $e*d = 1 \bmod \varphi(n)$;
5. (n, e) is the public key, and d is the private key.

The signature of message m is a straightforward modular exponentiation using the hash of the message and the private key;
The signatures can be obtained by:
$s = \text{hash}(m)\ d \pmod n$; [16]
To verify a signature for message m, the author's public key (n, e); is used to decrypt the signature. The hash h is thus obtained by $h = s^e \pmod n$; [16]
The signature is valid if h matches hash(m), then the message has not been altered and was signed by the author.

2.3 AES (Advanced Encryption Standard)

This is a renowned encryption algorithm that is widely adopted for secure data transmission and storage. It has been rigorously analyzed and is considered to be highly secure against cryptographic attacks. AES operates on fixed-size blocks of data and supports various key lengths such as 128, 192, and 256 bits. A substitution-permutation network (SPN) structure, consisting of several rounds of substitution, permutation, and mixing operations is employed by this algorithm [14]. AES encryption involves dividing the plaintext into blocks, which are then encrypted using the selected key and a series of transformations. Each round of encryption modifies the state of the data, making it resistant to statistical attacks and other cryptographic vulnerabilities. AES decryption follows a reverse process, using the same key to retrieve the original plaintext.

2.4 SHA (Secure Hash Algorithm)

To provide standardized cryptographic hash functions for widespread public use, the NIST (National Institute of Standards and Technology) selected a series of algorithms that were named SHA [8]. A procedure that returns a fixed-size bit string (hash value) for a block of data i.e. the message to be sent is known as a Cryptographic Hash Function [22]. An example of SHA is the SHA-256 which was used in this paper.

A 512-bit block size and a 256-bit key size are used in the 64 compression rounds of the SHA-256. The following steps are involved in SHA-256:

a. Padding: Length should be a multiple of 512 bits.

b. Initialization: Length is increased to 64 bits.
c. Processing: Each block undergoes four rounds of 20-bit operations and the message is processed in 512-bit blocks.
d. Hash value is the output.

A hash can be used to verify information that is too sensitive to store and it can act as a placeholder for a document. It can also be used to confirm that a document has not been tampered with [8]. Some of the properties hashes possess to be useful are: Two identical documents must yield different hashes and the original data must be nearly impossible to recreate from the hash. Calculating the hash should be executed swiftly and it should be impractical to create two sets of data that yield the same hash.
In cryptography, appending data to any part of the plaintext before encryption is known as padding. The purpose of padding is to prevent an adversary from retrieving information of the primitive, such as an adaptive chosen ciphertext attack in RSA [28].

3.0 Methodology

The methodology encompasses various techniques and approaches to ensure a thorough understanding of the system requirements and an effective design that meets the needs of the users.

3.1 Dataset Description

The data compared and analyzed in this paper were six (.txt) text files of varying sizes which are listed below:

i. 5 KB
ii. 10 KB
iii. 20 KB
iv. 30 KB
v. 50 KB
vi. 100 KB

Several metrics have also been identified to evaluate the performance of the Proposed System.

3.2 Performance Metrics

The Proposed System's performance was evaluated and analyzed against the AES algorithm using the following metrics:

- Execution time
  This is the total time it takes the system to carry out an encryption and decryption operation. It indicates the speed of the system.
- Size of the file
  This indicates if the process of signing the document affects the size of the document.

- Key size
  Different RSA keys were used to sign & verify the same document to determine the speed of the signature and verification operations.

3.3 Proposed System

The Python programming language was used to develop proposed system presented in this paper. It combines advantages of digital signatures alongside the strengths of symmetric encryption. The AES was used as the primary mode of encryption which was used to achieve confidentiality and authentication of the document while the RSA was used to implement the digital signature which ensures the integrity of the document and non-repudiation.

The sender encrypts the document using AES algorithm and also generates RSA Private and Public keys. The SHA-256 algorithm is used to hash the ciphertext to obtain a hash value. The digital signature is created by encrypting hash value using the private key of the RSA algorithm. The ciphertext and digital signature are attached and sent to the recipient.

After decrypting the digital signature to obtain the hash value of the ciphertext, the recipient verifies if it is a valid signature before proceeding to decrypt the ciphertext using the AES secret key as shown in figures 1, 2 and 3.

3.4 Architecture of the Proposed System

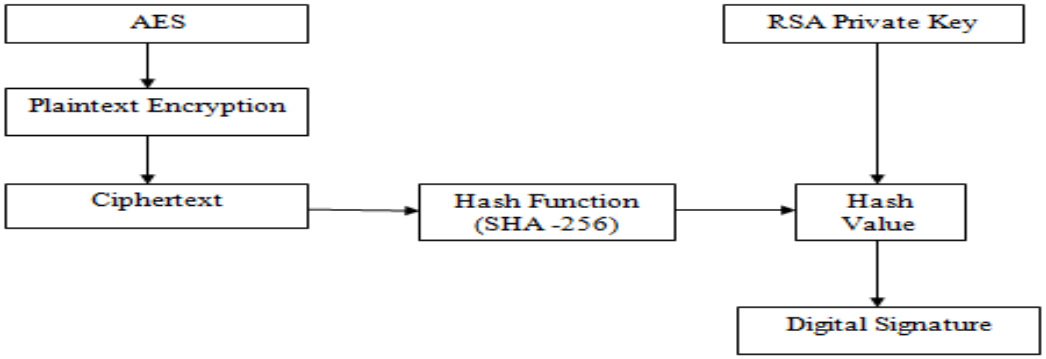

Figure 1: System Block Diagram

3.5 Flowchart of the Proposed System

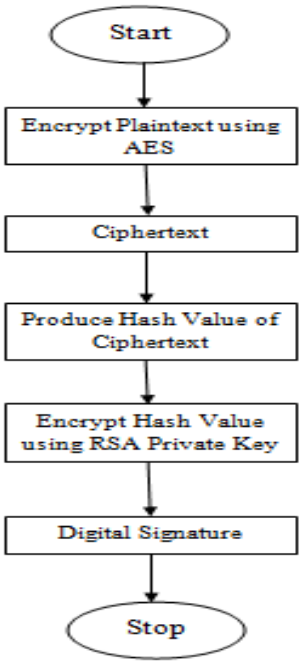

Figure 2: Encryption

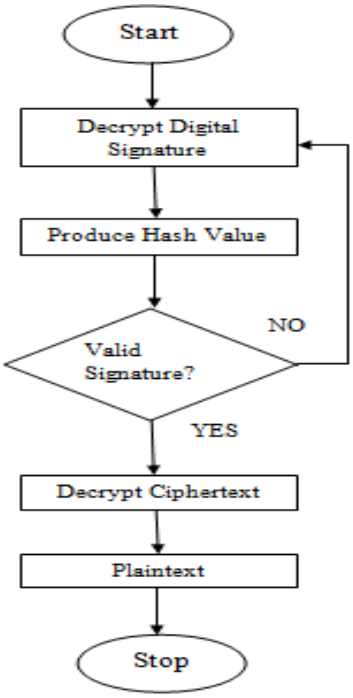

Figure 3: Decryption

4.0 Results

4.1 Performance Evaluation

The proposed systems' performance was evaluated against the AES algorithm and the results obtained were compared and evaluated.

Table 1: Execution Time of each System

File Size	AES Encryption	AES Decryption	Proposed System Encryption	Proposed System Decryption
5KB	0.39	0.97	0.39	0.97
10KB	1.15	5.07	1.16	5.07
20KB	2.17	8.38	2.17	8.38
30KB	3.42	13.22	3.42	13.22
50KB	6.16	20.30	6.17	20.30
100KB	11.35	45.41	11.35	45.41

Table 1 shows the processing times obtained from each algorithm and there is no significant change in processing times of the proposed system and the AES algorithm.

Table 2: Total Execution Time of each System

File Size	AES Total Time	Proposed System Total Time
5KB	1.36	1.36
10KB	6.22	6.23
20KB	10.55	10.55
30KB	16.64	16.64
50KB	26.46	26.47
100KB	56.76	56.76

Table 2 shows the total execution time of each algorithm. The execution times are almost identical and this shows that the digital signature signing and verification processes have no significant impact on the encryption and decryption time of the proposed system and also, as the file increases in size, the longer it took for the encryption and decryption processes.

4.2 Size of the File

The different files were observed and compared before encryption and after encryption to see if the proposed system affects the sizes of the files.

Table 3: File Size Comparison

File Size Before Encryption	File Size After Encryption
5KB	5KB
10KB	10KB
20KB	20KB
30KB	30KB
50KB	50KB
100KB	100KB

Table 3 shows that the proposed system did not reduce or increase the sizes of the various files used. The Digital Signature maintained the integrity of the files used.

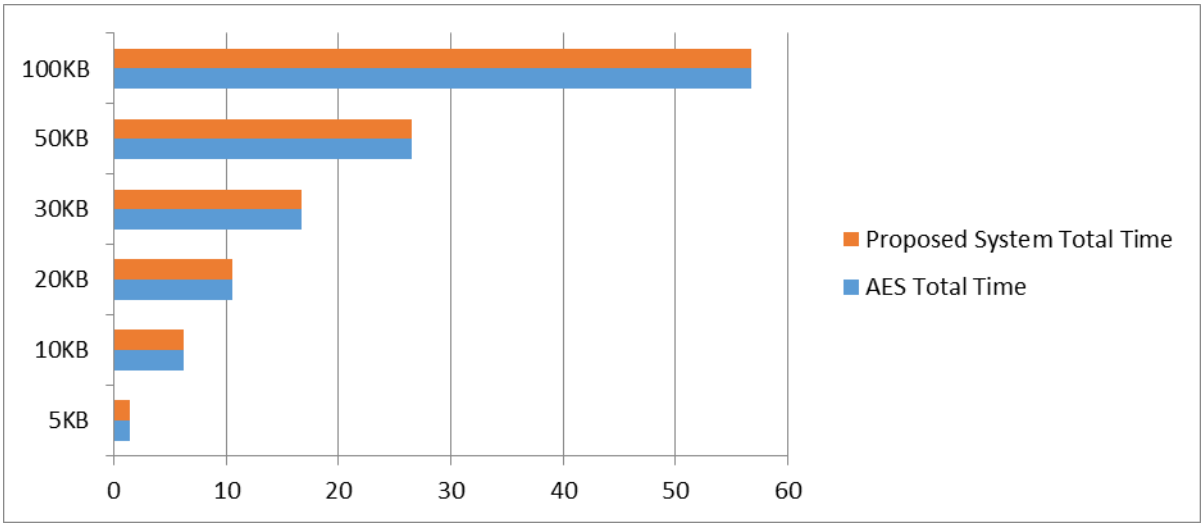

Figure 4: Total Execution Time of each System

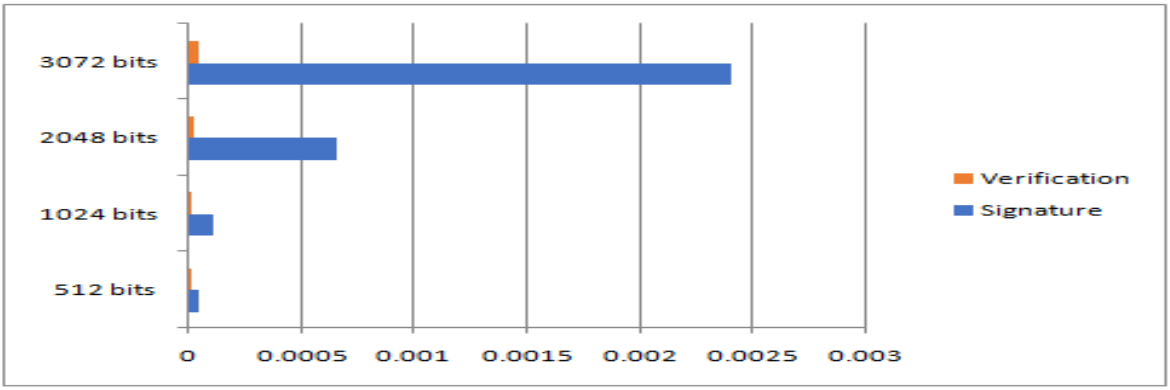

Figure 5: Signature and Verification time of different RSA key sizes

4.3 RSA Key Size

Different RSA key sizes were used to sign and verify the 100KB file to see how long each key takes to carry out signature and verification processes.

Table 4: RSA Key Sizes

RSA Key Size	Signature	Verification
512 bits	0.000043s	0.000002s
1024 bits	0.000105s	0.000008s
2048 bits	0.000656s	0.000020s
3072 bits	0.002395s	0.000043s

Table 4 shows the different key sizes and how they perform against each other. The smaller the key size the faster the signature and verification processes. The verification process is a much faster process than the signature process.

4.4 Discussion

The results obtained shows that the Digital Signature implemented in the proposed system does not affect the speed of the AES algorithm and the integrity of the document was maintained as it did not increase or reduce the sizes of the files after encryption as shown in figures 4 and 5. The proposed system is only able to encrypt and decrypt text (.txt) files as it was not designed to process other file types, this can be improved upon in future works that will be carried out.

In comparison to the work of Aufa, *et al.* [2] the proposed system uses RSA for digital signature and AES for symmetric encryption to achieve authentication and confidentiality of transmitted data whereas theirs uses RSA and DSA to achieve authentication and confidentiality of transmitted data. When compared to the work of William, *et al.* [25] their proposed system uses Elliptic curve cryptography (ECC) for digital signature operations and was more efficient for text encryptions and it could also process other file types.

5.0 Conclusion

The proposed system presented in this paper is seen to have met the four objectives of cryptography. The symmetric encryption achieves confidentiality and authentication while the digital signature achieves integrity and non-repudiation.

Further research should be carried out on how to implement digital signatures using other encryption algorithms and analyzing them against existing models. The proposed work can further be improved upon by addressing the issue of key management of symmetric encryption and upgrading the proposed system to process other file types.

References

[1] Anton, A., & Earp, J. (2021). Digital Signatures: Use Cases, Adoption Trends, and Opportunities. *Gartner Research.*

[2] Aufa, F. J., & Affandi, A. (2018). Security system analysis in combination method: RSA encryption and digital signature algorithm. *International Conference on Science and Technology (ICST) pp. 1-5*

[3] Bobby, M. & Usha, D. (2020). Observational Study of Security Enhancement in IoT Authentication Using Digital Signature Algorithm. *Journal of Xidian University. Vol. 14 (3). ISSN No:1001-2400*

[4] Carr, S. A., & Campbell, D. E. (2012). An investigation into the relationship between e-signature authentication and the intention to accept and use e-signatures. *Information & Management, 49(5), 218-228.*

[5] Dasttidar, S. G., & Mishra, A. (2023). Beyond Traditional Contracts: A Legal Analysis of Smart Contracts. *Issue 1 Indian JL & Legal Rsch., 5* (1).

[6] Dhiyaulhaq, D. H., & Usman, S, A. (2020). Comparative Performance of Digital Signature Security Using Cryptography AES 192 BIT and RSA 512 BIT Algorithm Model. *Journal of Advances in Information Systems and Technology 2(2) pp* 63-72

[7] Diffie, W., & Hellman, M. E. (1976). New Directions in Cryptography. *IEEE Transactions on Information Theory, 22(6), 644-654.*

[8] Dubrawsky, I. (2010) Eleventh Hour Security+. *Syngress. ISBN 9781597494274 Chapter 10, pp 135-151.*

[9] Garcia-Rodriguez, J., Moreno, R. T., Bernabe, J. B., & Skarmeta, A. (2021). Implementation and evaluation of a privacy-preserving distributed ABC scheme based on multi-signatures. *Journal of Information Security and Applications*, *62*, 102971.

[10] Gupta, A., Tung, Y. A., & Marsden, J. R. (2004). Digital signature: use and modification to achieve success in next generational e-business processes. *Information & Management, 41*(*5), 561-575.*

[11] Jamra, R. K., Anggorojati, B., Sensuse, D. I., & Suryono, R. R. (2020, October). Systematic Review of Issues and Solutions for Security in E-commerce. In *2020 International Conference on Electrical Engineering and Informatics (ICELTICs) (pp. 1-5). IEEE.*

[12] Jansma, N., & Arrendondo, B. (2004). Performance comparison of elliptic curve and rsa digital signatures.

[13] Lee, S., & Kim, J. (2019). Optimized RSA Encryption Techniques for File Encryption. *International Journal of Advanced Computer Science and Applications, 10(8), 264-270.*

[14] Lee, W. B., & Lee, C. D. (2008). A cryptographic key management solution for HIPAA privacy/security regulations. *IEEE Transactions on Information Technology in Biomedicine, 12(1), pp34-41.*

[15] Lu, Z., & Mohamed, H. (2021). A Complex Encryption System Design Implemented by AES. *Journal of Information Security. 12 (2).*

[16] Mao, W. (2003). Modern Cryptography: Theory and Practice. *Prentice Hall PTR. ISBN: 0130669431. pp. 258*

[17] Mougayar, W. (2016). The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology. *Wiley.*

[18] Pereira, C., Barbosa, L., Martins, J., & Borges, J. (2018). Digital signature solution for document management systems-the university of trás-os-montes and alto douro. In *Trends and Advances in Information Systems and Technologies:2 6*, pp *16-25. Springer International Publishing.*

[19] Refialy, L., Sediyono, E., & Setiawan, A. (2015). PengamananSertifikat Tanah Digital Menggunakan Digital Signature SHA-512 dan RSA. *Jurnal Teknik Informatika dan SistemInformasi 1 (3).*

[20] Rivest, R. L., & Dwork, C. (1997). Ad-hoc groups and the threat of unsynchronized signatures. *Advances in Cryptology—CRYPTO'97.*

[21] Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM. 21(2), 120-126.*

[22] Tsiatsis, V., Karnouskos, S., Höller, J., Boyle, D., & Mulligan, C. (2019). Internet of Things (Second Edition). *Academic Press. ISBN 9780128144350. Chapter 6, pp 127-142*

[23] Vijay, N., Singh, K., & Saxena, A. (2021). A Hybrid Digital Signature Technique using Cryptosystem. *International Journal of Computer Applications (0975 – 8887), 174(24).*

[24] Wali, A., Ravichandran, S., & Das, S. (2024). A 2D Cryptographic Hash Function Incorporating Homomorphic Encryption for Secure Digital Signatures. *Adv. Mater. 2024, 2400661.* *https://doi.org/10.1002/adma.202400661* [Accessed: 23-Oct-2024]

[25] William, P., Choubey, A., Chhabra, G.S., Bhattacharya, R., Vengatesan, K., & Choubey, S. (2022). Assessment of Hybrid Cryptographic Algorithm for Secure Sharing of Textual and Pictorial Content. *Proceedings of the International Conference on Electronics and Renewable Systems (ICEARS 2022) IEEE Xplore Part Number: CFP22AV8-ART; ISBN: 978-1-6654-8425-1*

[26] Xiao, F., & Chen, S. (2018). Security in the Internet of Things: A review. *IET Cyber-Physical Systems: Theory & Applications, 2(1), 13-27.*

[27] Yudistira, R. (2020). AES (Advanced Encryption Standard) and RSA (Rivest–Shamir–Adleman) Encryption on Digital Signature Document: A Literature Review. *International Journal of Information Technology and Business. 2(2), pp 26-29*

[28] Zhong, Y. (2022). An Overview of RSA and OAEP Padding. *International Conference on Computational Intelligence and Applications. Vol. 1*